**Homework: Indexing Exercises (2)**

Q1(a) Create an initial B+ tree with p = 3 and pleaf = 2

* + First Insert 8, 5,
  + Then insert the values in the following sequence
    - 1, 7, 3, 12, 9, 6
  + At each insertion observe, if it is increasing width or height of the tree. Observe how leaf and intermediate nodes split in case of overflow.

Q1(b) Deletion values from the tree you just created in the following sequence: 5, 12, 9.

Observe node merge and redistribution of values as well as pointer update.

You can use following visualization tool. However, also analyze manually to better understand the functionality of B + tree.

<https://www.cs.usfca.edu/~galles/visualization/Algorithms.html>

Q2(a): Use the same value from the question 1 and create a B tree using the visualization tool.

* Observe the differences of B tree with B+ tree and list down what make B tree different from B+ tree.
* In your opinion, why B tree is not suitable for DBMS indexing

Q2(b): Based on your observation, state what properties of B+ tree makes them popular DBMS technique.

*Note: Additional reading is available for Multi-Level, B and B+ trees, in the additional reading section under topic 2. Kindly refer to additional reading, while answering Q2.*

Q(3): Suppose we want to execute the following query:

Select \* From Student, Apply, College

Where Student.sID = Apply.sID and Apply.cName = College.cName

And Student.GPA > 1.5 And College.cName < 'Cornell'

Assume sID is unique. Create tree-based index (one at a time on) the following attribute pairs (Note: by default, indexes are tree-based).

Student.sID, College.cName

Student.sID, Student.GPA

Apply.cName, College.cName

Apply.sID, Student.GPA

Evaluate which two pair of the index is most useful for speeding up query execution. For each pair, discuss why or why not that index pair is useful based on the number of disk blocks required to access a record.